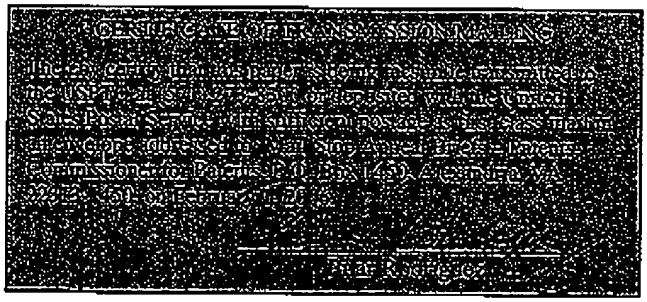
Patent

Attorney Docket No.: Intel 2207/10379
Serial No.: 09/811,111
Assignee: Intel Corporation

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICANT	:	Steve B. McGOWAN
SERIAL NO.	:	09/811,111
FILED	:	March 16, 2001
FOR	:	METHOD AND APPARATUS FOR ENCAPSULATING UNIVERSAL SERIAL BUS MESSAGING OVER LINK LAYER COMMUNICATION PROTOCOL
GROUP ART UNIT	:	2661
EXAMINER	:	Tri H. PHAN

M/S: APPEAL BRIEFS - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450



APPEAL BRIEF

Dear Sir:

This brief is in furtherance of the Notice of Appeal, filed in this case on September 1, 2005.

02/02/2006 CNGUYEN 00000024 110600 09811111
01 FC:1402 500.00 DA

81940.1

1. REAL PARTY IN INTEREST

The real party in interest in this matter is Intel Corporation. (Recorded March 16, 2001, Reel/Frame 011671/0133).

2. RELATED APPEALS AND INTERFERENCES

There are no related appeals.

3. STATUS OF THE CLAIMS

Claims 1-27 are pending in this application. Claims 1-27 are rejected. This appeal is an appeal from the rejection of claims 1-27.

4. STATUS OF AMENDMENTS

Applicant did not make any amendments to the claim subsequent to final rejection. The claims listed on page 1 of the Appendix attached to this Appeal Brief reflect the present status of the claims.

5. SUMMARY OF THE CLAIMED SUBJECT MATTER

The embodiment of claim 1 generally describes a system for communication between a host device and a peripheral device including a peripheral device (e.g., see page 4, line 14 – Figure 1, 104) to encode data and the host device (e.g., see page 4, line 13 – Figure 1, 102) to decode data under a Universal Serial Bus (USB) protocol (e.g., see page 4, line 8) to form a USB packet; wherein: the USB packet is encoded using a Bluetooth protocol to form a Bluetooth packet (e.g., see page 4, line 9) for the transmission between the host device and the peripheral device (e.g., see page 4, line 13-14).

The embodiment of claim 13 generally describes a method for communication between a host device and a peripheral device, comprising: encoding data under a Universal Serial Bus

(USB) protocol (e.g., see page 4, line 8) to form a USB packet; and encoding the USB packet with a Bluetooth protocol to form a Bluetooth packet (e.g., see page 4, line 9) for transmission between the host device and the peripheral device (e.g., see page 4, line 13-14).

The embodiment of claim 13 generally describes a system for communication between a host device and a Human Interface Device (HID) comprising: a peripheral device (e.g., see page 4, line 14 – Figure 1, 104) to encode data and the host device to decode data (e.g., see page 4, line 13 – Figure 1, 102) under an HID protocol to form a Universal Serial Bus (USB) packet; wherein: the USB packet is encoded using a Bluetooth protocol to form a Bluetooth packet (e.g., see page 8, line 7-8) for the transmission between the host device and the peripheral device by adding a transaction header to the USB packet (e.g., see page 8, lines 8-11) so that the USB packet is included as payload in the Bluetooth packet; a channel identifier (CID) is used to identify each endpoint of one or more endpoints associated to the peripheral device (e.g., see page 5, lines 1-2); the Bluetooth protocol utilizes a logical link control and adaptation protocol (L2CAP) to provide segmentation and reassembly (SAR) (e.g., see page 19, lines 15-20).

Figure 1 illustrates the communication flow between a host computer 102 and a peripheral device 104 according to an embodiment of the present invention. In this embodiment, a Bluetooth / HID protocol (BT-HID) provides a communication service between application software 106 and system software executed at a host computer and a peripheral device (BT-HID) device 104. The BT-HID device 104 can have different communication flow requirements for different application-to-device interactions. In this embodiment of the invention, good overall bus utilization is provided by allowing the separation of the different communication pathways 108 to a BT-HID device 104. Each communication pathway (or Pipe) 108 makes use of some bus access to accomplish communication between the application 106 and device 104.

Each pipe 108 is terminated at a logical channel endpoint 110 on the BT-HID device 104 at one end and a buffer 112 on the host 102 at the other end. Information associated with an endpoint 110 is used to identify the characteristics of the respective pipe 108. Channel IDs (CIDs) 114 are used to identify endpoints 110.

In an embodiment of the present invention, an HID device 104 appears to the system as a collection of endpoints 110. System software executed at host computer 102 manages the device using the Control Pipe 116. Application software 106 manages the device using the Interrupt Pipe 118 and the Control Pipe 116. Application software 106 requests that data be moved across the Bluetooth transmission between a buffer on the host 102 and an endpoint 110 on the BT-HID device 104. The host controller 120 (or BT-HID device controller 122 depending on transfer direction) packetizes the data to move it over the air. The host controller 120 also coordinates bus access to enable transferring the packet of data over the air.

Figure 2 illustrates an example of how payloads are translated between the data defined by the HID specification (top layer) and the baseband Bluetooth packets (bottom layer) and back. In an embodiment of the present invention, a BT-HID mini-driver 206 adapts the standard HID Parser and Transport 208 provided by an operating system to the Bluetooth stack 212,214. On the device 204 side, a generic HID Services module 210 provides the services required by any BT-HID implementation. HID Services 210 communicates through the L2CAP 212 and standard Bluetooth Control 214 interfaces. In this embodiment, the Application-Specific firmware 216 of a BT-HID device communicates with the HID Services 210, and provides the following functions: Enumeration; Authentication; Connection State Management; Segmentation and Reassembly (SAR); Packetization; and Scheduling.

Figure 3 illustrates the conversion of an HID data packet 304 to one or more Bluetooth baseband data packets 308. In one embodiment, a BT-HID device adds a header (also called the Transaction Header - THdr) 302 to the HID payload 304. A THdr 302 can vary in length from 1 to 3 bytes. The first byte of header 302 identifies the Transaction Type (TT) associated with the transaction.

In one embodiment, L2CAP adds another layer of encapsulation, defined by the Bluetooth Specification. In this L2CAP packet, the length 310 and the CID 312 is provided in the header. If the payload 306 resulting from the first (BT-HID) encapsulation is too large to fit in a single baseband packet 308, the L2CAP layer will segment the payload 306 into smaller blocks 308 before transmission or assemble segmented received packets into an L2CAP packet.

As stated, in an embodiment, all messages between an HID device and a host are preceded by a BT-HID Transaction Header (THdr) 302. In an embodiment, the Transaction Header 302 is divided into two (or more) fields: the transaction type and a transaction parameter (Also, a 'length' field can be added). The transaction parameter may be transaction type dependent.

Figure 4 illustrates the concatenation of the BT-HID payload according to an embodiment of the present invention. In the following discussion the term, 'Host' refers to the software stack on the host that supports BT-HID devices.

Figures 5a and 5b describe transfer from host to device and device to host, respectively, according to an embodiment of the present invention.

6. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

A. Are claims 1-3, 5, 13-15 and 17 anticipated by Trost et al. (U.S. Application No. 2002/0151275) (hereinafter "Trost")?

B. Are claims 4, 6-12, 16 and 18-27 obvious in view of Trost?

7. ARGUMENT

A. Claims 1-3, 5, 13-15 and 17 are not anticipated by Trost

Applicant respectfully submits that none of the cited sections of Trost teach, suggest or disclose at least "[a] system for communication between a host device and a peripheral device comprising: the peripheral device to encode data and the host device to decode data under a Universal Serial Bus (USB) protocol to form a USB packet; wherein: *the USB packet is encoded using a Bluetooth protocol to form a Bluetooth packet* for the transmission between the host device and the peripheral device" (e.g., as described in claim 1).

The Examiner asserts that forming the over-the-air packets for transmitting through the transceiver ("encoding USB packet into Bluetooth packet") is disclosed in Figure 13 and paragraphs [0086]-[0087] of page 6. Applicant disagrees. Paragraph [0086] of page 6 discloses:

> [0086] FIG. 13 is a graphical illustration of the layering and packets within the host and embedded software of a Bluetooth device. The upper most layer is the L2CAP layer 1301. An L2CAP packet produced by the L2CAP layer comprises a length field 1303, a channel ID 1305 and a payload 1307. The L2CAP packets are broken down into HCI packets in the HCI layer 1309. The HCI packets have flags to indicate whether they are the beginning of an L2CAP packet 1311 or a continuation of an L2CAP packet 1313. An HCI data payload 1315 will always end concurrently with an L2CAP payload 1307 in order to insure that the HCI packet does not straddle an L2CAP boundary. Accordingly a L2CAP packet always translates into an integer number of HCI packets.

Applicant respectfully submits the paragraph [0086] of Trost generally discloses:

1) the contents of a L2CAP packet and

2) the conversion of that L2CAP packet into a HCI packet.

Applicant submits this section does not disclose "[a] system for communication between a host device and a peripheral device comprising... wherein: *the USB packet is encoded using a Bluetooth protocol to form a Bluetooth packet* for the transmission between the host device and the peripheral device" as specifically described in the embodiment of claim 1.

Next, paragraph [0087] of Trost states:

> [0087] The next lower layer 1317 is indicated as a HCI-USB layer in the host. Layer 1317 transfers HCI packets to the USB driver 1321. The USB Driver layer 1321 then converts the HCI packets to USB packets and then transmits the USB packets, e.g. 1323A and 1325A, across a physical connection. The physical connection in the present examples indicated as a USB hardware bus, coupled to the Bluetooth device. The USB packets are received in a USB Driver layer. The USB packets are then reassembled into HCI packets 1319B within the Bluetooth device. The HCI packets 1319B will then be formed into the over-the-air packets 1331 which will then be sent out over the Bluetooth radio 1333. *An embodiment of the present invention inserts two more layers between the HCI packet 1319B and the over the air packet 1331. The first layer inserted is the layer that takes the HCI packet and forms firmware packet 1327. The second layer is a layer that maps the firmware packets 1327 into fragments 1329 within the transmit FIFO. The hardware will then automatically choose the over-the-air packets 1331 from the transmit FIFO.* The fragments belonging to each connection are treated as contiguous segments except for the marker which shows where the L2CAP boundary is. When an L2CAP boundary is encountered an optimally sized over-the-air packet that does not straddle an L2CAP boundary is selected and sent. *(emphasis added)*

Applicant respectfully submits paragraph [0087] of Trost are generally discloses:

1) the conversion of HCI packets into USB packets

2) the transmission of the USB packets across a physical connection

3) the reassembly of the USB packets into HCI packets within a Bluetooth device

4) the forming of the HCI packets into over-the-air packets, which are then sent over the Bluetooth radio.

Although the cited paragraph continues thereafter to describe insertion of layers into packets, these layers are inserted into HCI packets, *not USB* packets that are *encoded with a Bluetooth*

protocol to form a Bluetooth packet as specifically described in embodiments of the present application. Applicant submits that the two cited paragraphs [0086] and [0087] of Trost fail to disclose encoding a USB packet with Bluetooth protocol altogether.

Next, the Examiner cites paragraph [0041] as disclosing relevant limitations. Applicant disagrees. Paragraph [0041] states:

> [0041] FIG. 1 is a graphical representation of an exemplary Bluetooth environment. In FIG. 1 the following Bluetooth systems are illustrated. A Personal Digital Assistant (PDA) 103 is coupled to a Bluetooth wireless transceiver 105A. A fax machine 107 is coupled to a Bluetooth wireless transceiver 105B. A telephone 109 is coupled to a Bluetooth wireless transceiver 105C. A telephone network, represented by telephone wall plug 111, is coupled to a Bluetooth wireless transceiver 105D. A printer 113 is coupled to a Bluetooth wireless transceiver 105E. A computer is coupled to a Bluetooth transceiver 105F. A keyboard is coupled to a Bluetooth transceiver 105G. By using Bluetooth technology all of the devices of FIG. 1 can communicate with each using Bluetooth radio frequency (RF) connections without interconnecting cables.

Applicant submits that although the cited paragraph generally describes a various peripherals (e.g., a PDA or a fax machine) coupled to a Bluetooth wireless transceiver, there is no teaching, suggestion or disclosure of "[a] system for communication between a host device and a peripheral device comprising... wherein: *the USB packet is encoded using a Bluetooth protocol to form a Bluetooth packet* for the transmission between the host device and the peripheral device" as specifically recited in amended claim 1.

The Examiner also cites paragraphs [0070] and [0071] as disclosing the relevant limitations. Applicant disagrees. The cited paragraphs state:

> [0070] LMP packets are coupled between a link manager 705 and the physical layer 707. The link manager 705 is used to negotiate packet types between Bluetooth devices, as well as to setup encryption. The link manager 705 is also used for setting up SCO links and for other link management functions *FIG. 8 is a graphical illustration representing protocol layering, as might be found within a Bluetooth device coupled to a personal computer (PC) via a USB (Universal Serial Bus). The L2CAP protocol 803 will is executing in the PC. The L2CAP protocol 803 communicates with a host controller interface (HCI) 805. Host controller interface 805 will then communicate with a USB*

> *module 807. Bluetooth device is then coupled via USB layer 809 to the USB layer 807 within the computer. The USB within the Bluetooth device 809 then communicates with the host controller interface (HCI) 811, which in turn communicates with the physical layer 813 in the Bluetooth device. (emphasis added)*
>
> [0071] So, for example, something is typed on a keyboard. The input from the keyboard would be then converted into L2CAP packets in L2CAP layer 803. The L2CAP packets would be then communicated to the HCI layer 805 which would convert the L2CAP packets into HCI packets. The HCI packets would then be converted into USB packets in the USB layer 807 and coupled across the USB bus to the USB layer 809 within the Bluetooth device. The USB packets would then be reassembled into HCI packets in HCI layer 811, and then further coupled into the physical layer 813. Commonly the USB layer 809 and HCI layer 811 are implemented in firmware within the Bluetooth device.

Applicant submits, similar to the embodiments of Trost described above, the relevant sections of paragraph [0070] generally disclose:

a) an L2CAP protocol 803 executing within the PC

b) the L2CAP protocol 803 communicates with a host controller interface (HCI) layer 805

c) The HCI layer 805 then communicates with a USB module (layer 807)

d) USB layer 807 then communicates with USB layer 809 of a Bluetooth device

e) USB layer 809 communicates with the HCI layer 811 in the Bluetooth device

f) HCI layer 811 then communicates with the physical layer 813 in the Bluetooth device.

Similar to paragraph [0070], paragraph [0071] discloses an exemplary embodiment wherein L2CAP packets are converted to HCI packets, HCI packets are converted to USB packets, transmitted, converted again to HCI packets and finally transmitted along the physical layer. Applicant submits that these two sections fail to disclose "[a] system for communication between a host device and a peripheral device comprising... wherein: *the USB packet is encoded using a*

Bluetooth protocol to form a Bluetooth packet for the transmission between the host device and the peripheral device" as specifically recited in amended claim 1.

Next, the Office Action cites paragraphs [0089-0091]. Paragraph [0089] states:

> [0089] FIG. 15 is a graphical illustration of an embodiment of the invention. In FIG. 15 Bluetooth host 1501 comprises three layers. *The first layer 1503 is a higher layer such as the L2CAP layer. Just below the L2CAP layer 1503 is the HCI Driver layer 1505. Just below the HCI Driver layer 1505 is the physical bus 1507. The physical bus may be a variety of different type buses such as a USB, PCI bus, or another type of bus. The physical driver 1507 in the Bluetooth host then interfaces with the actual physical bus 1509. The physical bus 1509 is then coupled to the HCI firmware in the Bluetooth baseband device.* Although much of the HCI firmware functionality has been transferred into HCI hardware 1511, different embodiments may choose to keep different functions within the HCI firmware 1513. In addition, some residual HCI firmware 1513 may continue to run to interface with the link manager firmware 1515. Additionally another aspect of an embodiment of the invention is the ability to switch between hardware and firmware modes of operation. *(emphasis added)*

Applicant submits that paragraph [0089] discloses the three layers of a Bluetooth host – the L2CAP layer, the HCI driver layer and physical bus layer – and their interactivity. However, the cited section does not disclose "[a] system for communication between a host device and a peripheral device comprising... wherein: *the USB packet is encoded using a Bluetooth protocol to form a Bluetooth packet* for the transmission between the host device and the peripheral device" anywhere.

Paragraph [0090] states:

> [0090] L2CAP data travels from the physical bus hardware 1509 through coupling 1519 into the HCI hardware 1511. The baseband controller block 1517 represents the physical layer. The physical layer couples the data into the Bluetooth RF (radio frequency) 1523, which in turn provides the data to an antenna 1525, which transmits the data over the air.

Paragraph [0090] discloses the transmission of the L2CAP data to the physical layer (by Bluetooth RF), and on to its final destination the antenna 1525. It does not disclose "[a] system for communication between a host device and a peripheral device comprising... wherein: *the*

USB packet is encoded using a Bluetooth protocol to form a Bluetooth packet for the transmission between the host device and the peripheral device" anywhere.

Finally, paragraph [0091] states:

> [0091] FIG. 16 is a high level block diagram of an exemplary baseband implementation. 1603 and 1611 comprise the link manager, which is in firmware. UART and USB 1605 may be firmware drivers handling the USB and UART functions. In certain embodiments of the invention the USB and UART may be done in hardware, UART/USB 1605 might go directly to a UART or USB port without first going through firmware drivers. In the case where the L2CAP packets are handled in hardware UART/USB 1605 would comprise physical UART/USB transmit and receive FIFOs. 1607 is a PCM (Pulse Code Modulation) interface. The PCM interface 1607 may be an interface to a codec for example in order to handle SCO traffic. Paths 1613 and 1623 are used for LMP packet traffic, paths 1615 and 1621 are for L2CAP traffic and paths 1617 and 1619 are for the SCO traffic. The LMP transmit FIFO is illustrated at 1625. The L2CAP transmit FIFO is illustrated at 1627, and the SCO transmit FIFO is illustrated at 1629. The SCO controller 1631 controls the SCO receive and transmit, FIFOs. As well as SCO receive and transmit timing. The SCO receive FIFO is illustrated at 1633, the L2CAP receive FIFO is illustrated at 1635, and the LMP receive FIFO is illustrated at 1637. Audio processor block 1641 handles PCM, Mu-Law and A-Law voice algorithms.

The cited paragraph describes a baseband implementation utilizing L2CAP packets. There is no mention of the use of Bluetooth protocol anywhere in the paragraph.

Therefore, Applicant respectfully submits that none of the cited paragraphs teach, suggest or disclose "[a] system for communication between a host device and a peripheral device comprising... wherein: *the USB packet is encoded using a Bluetooth protocol to form a Bluetooth packet* for the transmission between the host device and the peripheral device" as specifically recited in amended claim 1. Support can be found in the description of Figure 3 (please see pages 7 and 8 of specification)[1]:

> Figure 3 illustrates the conversion of an HID data packet 304 to one or more Bluetooth baseband data packets 308. *In one embodiment, a BT-HID device adds a header (also called the Transaction Header - THdr) 302 to the HID payload 304.* A THdr 302 can

[1] Applicant submit that the support is provided for clarification purposes only. Applicant maintains the cited references do not disclose the relevant claimed limitations found in embodiments of the present application.

> vary in length from 1 to 3 bytes. The first byte of header 302 identifies the Transaction Type (TT) associated with the transaction.
>
> *In one embodiment, L2CAP adds another layer of encapsulation, defined by the Bluetooth Specification.* In this L2CAP packet, the length 310 and the CID 312 is provided in the header. If the payload 306 resulting from the first (BT-HID) encapsulation is too large to fit in a single baseband packet 308, the L2CAP layer will segment the payload 306 into smaller blocks 308 before transmission or assemble segmented received packets into an L2CAP packet.

As shown above, these limitations are not described anywhere in the Trost reference. Therefore, Applicant submits since each and every limitation of independent claim 1 is not present in the cited references, the 102(e) rejection should be withdrawn. Independent claims 13 and 25 contain similar limitations to independent claim 1, and therefore are in condition for allowance as well. Claims 2-12, 14-24 and 26-27 are allowable for depending from allowable base claims.

B. Claims 4, 6-12, 16 and 18-27 are not obvious under Trost

Claims 4, 6-12, 16 and 18-27 were rejected under 35 U.S.C. § 103(a) as being unpatentable over Trost. Applicant respectfully submits, claims 4, 6-12, 16 and 18-27 are allowable as depending from an allowable base claim (see above). Based on the arguments above, reconsideration and withdrawal of the rejection of claims 4, 6-12, 16 and 18-27 under 35 U.S.C. §103(a) is respectfully requested.

Serial No. 09/811,111
Appeal Brief Filed February 1, 2006
Advisory Action dated September 23, 2005

CONCLUSION

Appellants therefore respectfully request that the Board of Patent Appeals and Interferences reverse the Examiner's decision rejecting claims 1-27 and direct the Examiner to pass the case to issue.

The Examiner is hereby authorized to charge the Appeal Brief fee of $500.00 and any additional fees which may be necessary for consideration of this paper to Kenyon & Kenyon Deposit Account No. 11-0600.

Respectfully submitted,
KENYON & KENYON

Date: February 1, 2006

By: _____
Sumit Bhattacharya
(Reg. No. 51,469)

KENYON & KENYON
333 West San Carlos St., Suite 600
San Jose, CA 95110
Telephone: (408) 975-7500
Facsimile: (408) 975-7501

Customer No. 25693

APPENDIX

(Brief of Appellants Steve B. McGowan
U.S. Patent Application Serial No. 09/811,111)

8. CLAIMS ON APPEAL

1. (Previously presented) A system for communication between a host device and a peripheral device comprising:

the peripheral device to encode data and the host device to decode data under a Universal Serial Bus (USB) protocol to form a USB packet;

wherein:

the USB packet is encoded using a Bluetooth protocol to form a Bluetooth packet for the transmission between the host device and the peripheral device.

2. (Original) The system of claim 1, wherein the USB packet is encoded using the Bluetooth protocol by adding a transaction header to the USB packet so that the USB packet is included as payload in the Bluetooth packet.

3. (Original) The system of claim 2, wherein the peripheral device is a Human Interface Device (HID).

4. (Original) The system of claim 3, wherein the USB protocol is an HID protocol.

5. (Original) The system of claim 2, wherein a channel identifier (CID) is used to identify each endpoint of one or more endpoints associated to the peripheral device.

6. (Original) The system of claim 2, wherein the Bluetooth protocol utilizes a logical link control and adaptation protocol (L2CAP) to provide segmentation and reassembly (SAR).

7. (Original) The system of claim 6, wherein the Bluetooth packet is encapsulated into a L2CAP packet of a packet size in preparation for conversion to the one or more baseband packets for Bluetooth transmission.

8. (Original) The system of claim 7, wherein the Bluetooth protocol utilizes the L2CAP to provide SAR in the conversion of the L2CAP packet to the one or more baseband packets when the packet size is too large to include the information of the L2CAP packet in one baseband packet of the one or more baseband packets.

9. (Original) The system of claim 8, wherein the Bluetooth protocol utilizes the L2CAP to provide SAR when the packet size is larger than a maximum transmission unit of each baseband packet of the one or more baseband packets.

10. (Previously presented) The system of claim 9, wherein the one or more baseband packet is capable of being transmitted from the host to the HID and from the HID to the host.

11. (Original) The system of claim 10, wherein upon transmission from the host to the HID, the HID is capable of recognizing any among a timeout, a data signal, or a stall signal.

12. (Previously presented) The system of claim 11, wherein upon transmission from the HID to the host, the host is capable of recognizing any among the timeout, an acknowledgement signal, a non-acknowledgement signal, or the stall signal.

13. (Original) A method for communication between a host device and a peripheral device, comprising the steps of:

encoding data under a Universal Serial Bus (USB) protocol to form a USB packet; and

encoding the USB packet with a Bluetooth protocol to form a Bluetooth packet for transmission between the host device and the peripheral device.

14. (Original) The method of claim 13, wherein the USB packet is encoded using the Bluetooth protocol by adding a transaction header to the USB packet so that the USB packet is included as payload in the Bluetooth packet.

15. (Original) The method of claim 14, wherein the peripheral device is a Human Interface Device (HID).

16. (Original) The method of claim 15, wherein the USB protocol is an HID protocol.

17. (Original) The method of claim 14, wherein a channel identifier (CID) is used to identify each endpoint of one or more endpoints associated to the peripheral device.

18. (Original) The method of claim 14, wherein the Bluetooth protocol utilizes a logical link control and adaptation protocol (L2CAP) to provide segmentation and reassembly (SAR).

19. (Original) The method of claim 18, wherein the Bluetooth packet is encapsulated into a L2CAP packet of a packet size in preparation for conversion to the one or more baseband packets for Bluetooth transmission.

20. (Original) The method of claim 19, wherein the USB/Bluetooth protocol utilizes the L2CAP to provide SAR in the conversion of the L2CAP packet to the one or more baseband packets when the packet size is too large to include the information of the L2CAP packet in one baseband packet of the one or more baseband packets.

21. (Original) The method of claim 20, wherein the Bluetooth protocol utilizes the L2CAP to provide SAR when the packet size is larger than a maximum transmission unit of each baseband packet of the one or more baseband packets.

22. (Previously presented) The method of claim 21, wherein the one or more baseband packet is capable of being transmitted from the host to the HID and from the HID to the host.

23. (Original) The method of claim 22, wherein upon transmission from the host to the HID, the HID is capable of recognizing any among a timeout, a data signal, or a stall signal.

24. (Previously presented) The method of claim 23, wherein upon transmission from the HID to the host, the host is capable of recognizing any among the timeout, an acknowledgement signal, a non-acknowledgement signal, or the stall signal.

25. (Previously presented) A system for communication between a host device and a Human Interface Device (HID) comprising:

a peripheral device to encode data and the host device to decode data under an HID protocol to form a Universal Serial Bus (USB) packet;

wherein:

the USB packet is encoded using a Bluetooth protocol to form a Bluetooth packet for the transmission between the host device and the peripheral device by adding a transaction header to the USB packet so that the USB packet is included as payload in the Bluetooth packet;

a channel identifier (CID) is used to identify each endpoint of one or more endpoints associated to the peripheral device;

the Bluetooth protocol utilizes a logical link control and adaptation protocol (L2CAP) to provide segmentation and reassembly (SAR).

26. (Original) The system of claim 25, wherein the Bluetooth packet is encapsulated into a L2CAP packet of a packet size in preparation for conversion to the one or more baseband packets for Bluetooth transmission.

27. (Original) The system of claim 26, wherein the Bluetooth protocol utilizes the L2CAP to provide SAR when the packet size is larger than a maximum transmission unit of each baseband packet of the one or more baseband packets.

9. EVIDENCE APPENDIX

No further evidence has been submitted with this Appeal Brief.

10. RELATED PROCEEDINGS APPENDIX

Per Section 2 above, there are no related proceedings to the present Appeal.